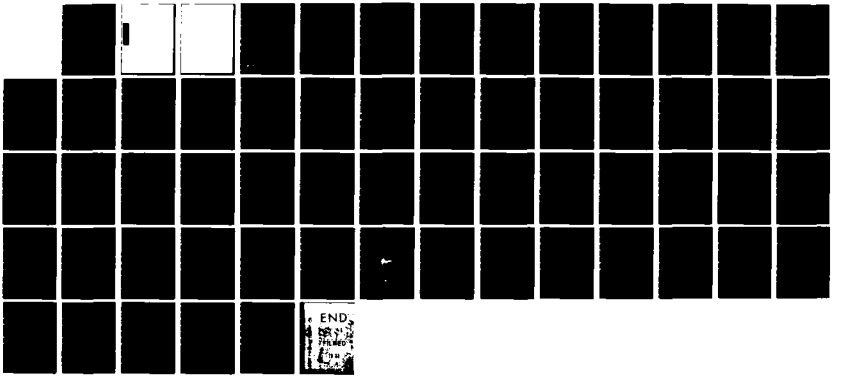
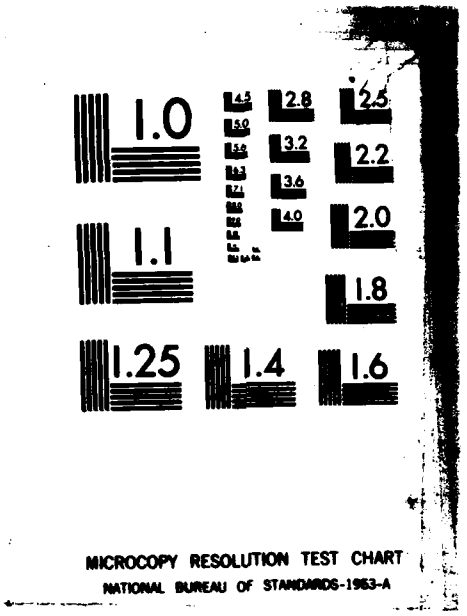
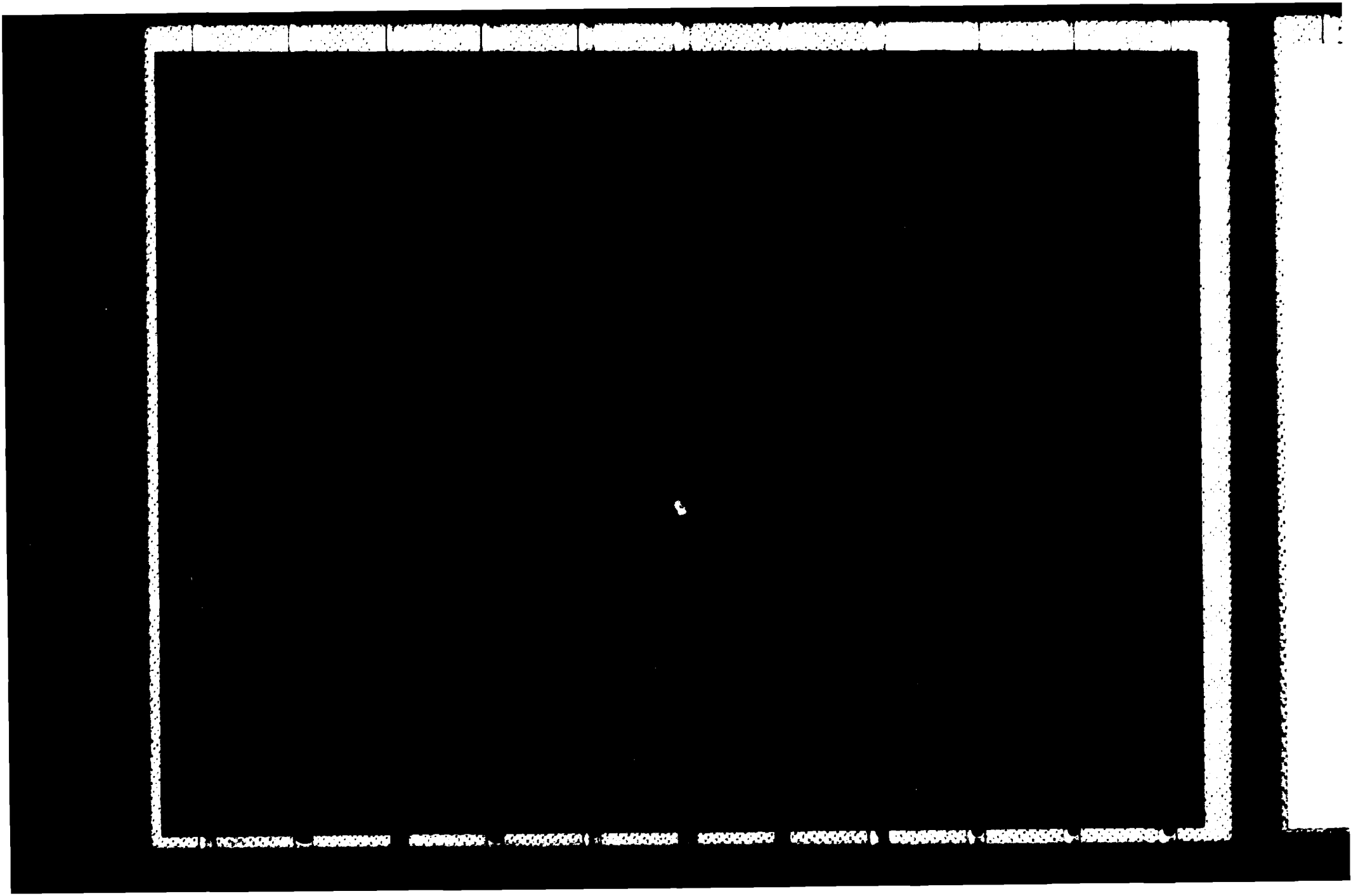
AD-A133 404 PERSPECTIVES ON OVERSIGHT MANAGEMENT OF SOFTWARE DEVELOPMENT PROJECTS(U) RAND CORP SANTA MONICA CA W H WARE ET AL. JUL 83 RAND/N-2027-AF F49620-82-C-0018 1/1

UNCLASSIFIED F/G 5/1 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A133404



A RAND NOTE

PERSPECTIVES ON
OVERSIGHT MANAGEMENT OF
SOFTWARE DEVELOPMENT PROJECTS

Willis H. Ware, Robert L. Patrick

July 1983

N-2027-AF

Prepared for

The United States Air Force

DTIC
ELECTE
OCT 1 2 1983
S D
A

35th Year

Rand
SANTA MONICA, CA. 90406

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

NT NUMBER: N-2027-AF	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): Perspectives on Oversight Management of Software Development Projects	5. TYPE OF REPORT & PERIOD COVERED: Interim
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Willis H. Ware, Robert L. Patrick	8. CONTRACT OR GRANT NUMBER(s): F49620-82-C-0018
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Rand Corporation, 1700 Main Street, Santa Monica, CA. 90406	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Requirements, Programs & Studies Group (AF/RDQM), Ofc, DCS/R&D and Acquisition, Hq, USAF, Washington, DC 20330	12. REPORT DATE: July 1983
	13. NUMBER OF PAGES: 50
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

No Restrictions

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Data Management
Air Force
Systems Management

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See Reverse Side

This report examines

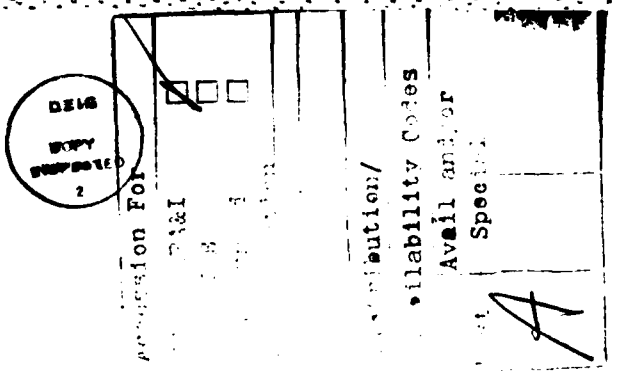
An examination of the problems of oversight management of large, important software development projects, i.e., management at levels above that of direct project execution. Senior managers must recognize events in the evolution of the project that might provide high leverage for maximizing the return on their invested time (in knowledge, status, or outcome projections), and that provide insight into project status and clues about possible problems. The Note attempts to identify the unique needs of ovesight managers, to indicate how they have been met in industrial environments, to enumerate what working supervisors and project managers must do differently if they are involved with large projects requiring oversght management, and to list what must be done (generally) to build software tools that supply information for oversight management as a by-product of the development process. While the discussion is based on industrial experience, it is also relevant to Air Force oversight management and review.

PREFACE

This Note was prepared for a summer study workshop sponsored by the Air Force Systems Command (AFSC) in July 1983 to address the software development process. It resulted from the authors' observation that the software management process at levels above the development project level has received little attention in the literature or in the research community. This work was produced as a part of the Project AIR FORCE administration.

The Note, which is based primarily on the authors' extensive experience in the design and implementation of complex software systems in the Air Force, the Department of Defense, and industry, is intended to support the summer study. It gives particular emphasis to the role and techniques of "oversight management" in helping to assure successful completion of software developments. The ideas discussed here should be of interest to any level of management within the Air Force that must review and monitor software-intensive projects. They should be of particular value to the Assistant Secretary of the Air Force for Financial Management and his staff; the Air Staff, especially the Assistant Chief of Staff for Information Systems and the Office of the Deputy Chief of Staff of the Air Force for Research, Development and Acquisition (AFRD); AFSC and its product divisions; the Air Force Logistics Command (AFLC) and its Air Logistics Centers; unified and specified commands and their data automation organizations; major commands and their data automation organizations; functional area centers such as the Air Force Military Personnel Center and the Air Force Finance Center; and the Air Force Communications Command (AFCC) plus its Data Systems Design Center and Communications Computer Programming Center.

A companion Note entitled "Perspectives on Life-Cycle Support of Software" is presently in preparation.



SUMMARY

This Note examines the oversight management of software-intensive projects, i.e., management at levels above that of project management, both within the development organization and in the client organization.

Large industrial firms have developed approaches to oversight management with varying degrees of success. The successful approaches appear to be based on four fundamental axioms:

- **A proper plan is the key to oversight management.**
- **Management's information needs differ at each level.**
- **Specialized reporting to oversight management diverts project personnel.**
- **Data for oversight management must have assured integrity.**

Case histories of software-intensive development efforts suggest various techniques for the conduct of oversight management and lead to principles that can be adapted for Air Force use:

- **The meaning of information for oversight managers must be easy to assimilate upon initial introduction.**
- **The relearning time during subsequent presentations of information must be minimal.**
- **The overall responsibility of oversight management is to promote integrity and forthrightness in the conduct of the project.**
- **Information provided to oversight managers must be organized and presented in a way that is appropriate for the managers' backgrounds, experience, willingness to contribute, and ability to understand.**
- **The review function provided by oversight management should continue throughout the lifetime of a project, although the periodicity of the meetings can and should change to match the pace, and possibly the stage, of the endeavor.**

Oversight managers of very large or very important software development projects must be able to recognize the events in the development process that provide high leverage for creating awareness of program status and that maximize return on their invested time--in terms of knowledge, status, or outcome projections. The Note identifies unique information needs of oversight managers, indicates how they have been successfully met in industrial environments, discusses ways in which the working supervisor and the project manager can provide appropriate information for oversight management, and presents some general suggestions that could lead to the development of software tools that supply information for oversight management as a natural by-product of the development process.

Fifteen proven oversight techniques are presented and related to a variety of decisions that oversight management might have to make. Finally, a series of Air Force actions and possible R&D efforts are suggested.

ACKNOWLEDGMENTS

Only four weeks elapsed between the first discussion about producing this Note and final camera-ready copy. Important to such a fast schedule was the secretarial support of Mel Carlson and the editorial and on-line wizardry of Janet DeLand.

CONTENTS

PREFACE iii

SUMMARY v

ACKNOWLEDGMENTS vii

Section
I. INTRODUCTION 1

II. DUTIES AND RESPONSIBILITIES OF AN OVERSIGHT MANAGER 8

III. AXIOMS OF OVERSIGHT MANAGEMENT 12
1. A Proper Plan is the Key to Oversight Management 12
2. Management's Information Needs Differ at Each Level 14
3. Specialized Reporting to Oversight Management Diverts Project Personnel 14
4. Data for Oversight Management Must Have Assured Integrity 18

IV. PRINCIPLES FROM CASE HISTORIES 20
Case 1 20
Case 2 24
Case 3 25
Case 4 28

V. A MENU OF PROVEN OVERSIGHT MANAGEMENT TECHNIQUES 30

VI. R&D POSSIBILITIES 44

Appendix: PROBABILISTIC BUDGETING: A PROMISING BUT UNTESTED METHOD 49

I. INTRODUCTION

Despite the explosive growth of computer applications over the past three decades, software management methods have been remarkably resistant to change. In the late 1940s and early 1950s, one individual frequently served as the programmer, the analyst, the mathematician, and the operator; in most cases, he was also the user. In this early period, each program was an end unto itself, directly benefiting the developer and/or his immediate colleagues. Management consisted of words of encouragement from senior colleagues and an occasional offer of help--but little participation--on the part of upper-echelon managers, most of whom were put off by the new technology. In the mid-1950s, computer manufacturers discovered that software would sell machines. Operating systems were introduced that made machines more convenient and efficient to use. Gradually, utility programs were also developed to facilitate programming; and eventually, high-order programming languages appeared. The vendors had to create project management strategies to develop hardware and software on compatible schedules, and these strategies continue to evolve as manufacturers attempt to provide full-function, tested software on schedules set by the pressures of the marketplace.

The early organizations had formally distinct groups or departments for marketing, hardware development, and software development. To solve the communication problems and resolve the conflicting interests among these organizational units, the project office was created. In this same era, several large computer-based military projects (primarily in command and control) were established, and these had similar internal structures for project management and control. Both the commercial manufacturers and the command-and-control projects had levels of management above the project, but this higher-level management consisted largely of reviewing monthly reports from the project manager and attending periodic status briefings on the schedule, the finances, and the general well-being of the effort. The senior managers were seldom computer

literate,[1] and the project manager controlled the agenda for these early executive briefings.

Between the late 1950s and the early 1970s, a quiet evolution in software project management took place. Talented administrators adapted management tools that had proved useful in other environments and developed innovative tools to fit the special needs of computing. PERT (program evaluation and review technique) charts, project schedules, and accounting systems that collected project costs and organized them for historical reference came and went. Trade publications and computer conferences invariably included "How To" presentations that made the job of the working line supervisor easier or supplied information for the project manager on how to keep the funds flowing and how to protect workers and their immediate management from undue schedule pressures.

The high point in this evolution occurred in the mid-1970s when the term "software engineering" was coined to describe the management tools and discipline required to bring additional order to the development process. However, the world was not standing still. The development of more reliable hardware, faster computers, larger memories, and dependable bulk storage devices allowed planners (both military and civilian) to conceive of computer-based systems orders of magnitude larger and more complex than any that had ever been built. While the difficulty of successfully managing such projects also increased, managers had had a decade or so of experience with software projects, and significant R&D attention had turned to tools and techniques for project managers.

As computer use grew, so did the problems of senior management. Not only were computer developments themselves very expensive, but computers and their software had become so embedded in large systems

---

[1] "Computer literate" is a loosely defined term implying an awareness on some level of understanding about aspects of computer affairs. It is usually applied to individuals who are not formally trained in a computer discipline but who need an overview, often in depth, of the field for some purpose. From the viewpoint of the professional practitioner, "computer literate" would describe the lay person who can speak knowledgeably about computers in general. In this Note, the term emphasizes the insights and perspectives relevant to overseeing computer-related project development.

(both civilian and military) that one system was often worthless without reliable operation by another. A modern commercial airline, for example, could not function without a reliable computer reservation system. Without a computer-based ground cart to diagnose on-board electronics and load the mission profile, the airborne capability of modern military aircraft with sophisticated avionics would rapidly degrade.

The computer is unquestionably in the mainline. The mission does not fly, the missile cannot be targeted, and the war games cannot be played without the computer component properly functioning. Unfortunately, not nearly as much research attention has been directed toward providing higher-echelon managers with the information necessary to track a software project, evaluate its state of health, or make crucial decisions about it.

Complex computer system development requires a well-organized management review process. Management at levels above that of direct project management is essential to the success of any large software development venture. While the project office and the system project manager watch progress, define problems, and arrange for solutions to problems throughout the development, there must be an oversight manager who can understand the status of the project without being buried by the detail and who can concern himself with larger issues such as multiyear funding, scheduling for initial operation, producing a trained staff, taking delivery of production systems, and integrating the new capability into the present environment.

There are at least two, and sometimes more, levels of line management on large efforts. The lowest level of line management, the working supervisor, spends 100 percent of his workday on the project; he works only on technical activities. He may be the leader of a design team, a programming team, a testing team, or an inspection team. He may produce documentation, package programs for delivery, determine requirements, or design systems. While he may occasionally attend meetings, prepare estimates and budgets, or read background material, his basic

responsibilities are decidedly technical and primarily concerned with a particular function to be achieved in the final system.

There may be several echelons of supervisory management. At the higher levels, project managers split their time between technical and administrative activities. They work extensively on budgets, status reports, and reviews of documentation, and they coordinate various aspects of the project. They attend many meetings that may have only moderate technical content. At the beginning of a project, they are deeply involved in the technical aspects of requirements, systems analysis, and the architecture of the final system. As the project moves forward, their duties become less technical and they become more involved in schedules, budgets, and staffing. At the top level are senior managers who have a direct interest in the outcome of the project.

Above the project level are oversight managers within the same development organization as the project. They carry the ultimate authority and can stop or suspend a project or change its basic direction, but they wield this authority sparingly. An oversight manager is interested in the project's outcome, but he must also be concerned with many other aspects: Is the project on schedule? Is it within budget? Will the desired functional capability be delivered at initial system deployment? Are there problems on the horizon that could have a significant effect on these projections? Are there danger signals that should be heeded to avoid undesired political alternatives or consequences? Finally, if the project is for an outside client, e.g., the Air Force, there will be one or more levels of oversight attention from the client's point of view.

We can illustrate the situation with a hypothetical major Air Force project being managed through one of the product divisions of the Air Force Systems Command (AFSC), say, the Aeronautical Systems Division (ASD). Typically, a program management office (PMO) or a system project office (SPO) is responsible for managing the effort from the Air Force side. This office is generally concerned with schedule, budget, and capability. Within it are specialists, engineers, and other skilled personnel. The SPO may also have a software specialist, but his experience may not include monitoring software development efforts.

Oversight reviews begin at ASD. They consider schedule, budget, and capability, but they are primarily concerned with problems, slippage, etc. Next, there is an AFSC-level review. Concerns at this level are probably similar to those at the ASD level, but more emphasis is placed on total program cost, because AFSC must justify budget projections to HqUSAF and eventually to Congress.

The next step is the Air Staff review. This is likely to approximate the AFSC-level review, but with even stronger emphasis on program cost and schedule; there is still some concern with capability but little if any concern with technical details, unless one of those details happens to be the cause of some schedule or cost problem. Air Staff-level review can involve a variety of general officers. At a minimum, the Office of the Deputy Chief of Staff of the Air Force for Research, Development and Acquisition (AFRD) will be involved, probably to the three-star level for large programs. One or more executives from the Chief of Staff's office may also be involved in the case of large programs or special reviews. There are likely to be general officers or senior civilians from the Comptroller's office because of funding considerations. Finally, large programs and periodic (e.g., annual) total Air Force budget program presentations to Congress may undergo a Secretary of the Air Force review.

Issues at yet a higher level of aggregation that deal with all the services may go to a Secretary of Defense review. Here, concerns are focused on cost and schedule issues, because of the interface to Congress and other parts of the Executive Branch of the government.

Thus, concern with technical details and with analytic examination of them decreases steadily with ascending levels of review. Technical details may be discussed at any level, but for the most part, decisions are based on detailed information provided by the SPO, perhaps supplemented with contributions from AFSC or the Air Staff. Similarly, the individuals who participate in the higher levels of review tend to be less technically and more globally oriented. They may be equipped to understand and discuss technical details, but it is unlikely that significant analytic examination will be done by anyone above SPO or possibly ASD (in this example) level. This is particularly true in matters of

computer hardware or software, simply because of the limited experience of the Air Force officer corps and civilian managers with these topics.

The oversight manager primarily looks outward, while the project manager looks into the project activities; a client's project office falls between them. The oversight manager is concerned about how the project is likely to affect, or be accepted by, the outside world. Thus, he has a "marketing" or advocacy orientation.

Table 1 illustrates and contrasts typical activities of the three levels of management in an industrial situation.

Table 1

TYPICAL MANAGEMENT ACTIVITIES

Management Level	Focus of Attention (percent of work time)		
	Technical Activities	Administrative Activities	Marketing Activities
Line management	100	0	0
Project management	50	50	0
Oversight management[a]	1	6	3

[a]The oversight manager spends only a small fraction of his time on any one project. For the Air Force, "oversight management" should be interpreted as a collective term for all such levels of management that might be involved.

Clearly, the oversight manager has different information requirements from the line or project managers; and he must be able to evaluate the data he receives in the proper context.

The boundary between project and oversight management is obviously fuzzy; in a sense, a PMO or SPO does both. However, the people assigned to these offices are closely connected to the effort, have a continuous relation to it, and are generally well informed on technical details. In contrast, oversight managers have intermittent contact with the project, need a periodic overview of it, must develop clues and insights about project affairs, and generally face away from the project toward such things as funding, advocacy of the program, reassurances to the

political structure, major program decisions, and interaction with end-user organizations.

The oversight manager's interest in the project and his need for project details vary, depending on his level in the management hierarchy. His needs for information about the project and information derived from it also vary. The obvious concerns of oversight review are functional capability, funds, and schedule: Will the project produce what is needed on schedule and within the budget? Yet there is a much broader set of issues for which an oversight manager may need information, both before and during a project. The techniques described in subsequent sections of this Note can provide information for such oversight functions as

- Initial project approval
- Selection of contractors
- Selection of key people
- Project estimating
- Monitoring of project schedule
- Monitoring of project funding
- Monitoring of technical details
- Monitoring of functional capabilities
- Monitoring of skill matches of key people and jobs
- Monitoring of schedule and funding by project phase
- Monitoring of project quality vs. original project requirements
- Monitoring progress of particular phases (e.g., design, programming, testing)
- Project reorganization
- Project redirection

The following sections describe the duties and responsibilities of oversight management and suggest some axioms that experience has shown are fundamental to good oversight. Four case histories are used to develop relevant principles for Air Force oversight management. Finally, a menu of proven oversight techniques is presented and related to the various tasks of such management. From it we derive several R&D possibilities.

II. DUTIES AND RESPONSIBILITIES OF AN OVERSIGHT MANAGER

Oversight management typically must accomplish five things:

1. Track the status of the project in terms of schedule, budget, and functional capabilities.
2. Assure that some knowledgeable person, not associated with the project, reads every formal document the project produces and questions any gross omissions or unworkable assumptions.
3. Assure that the project as currently defined and scheduled fulfills a necessary mission and is cost-effective.
4. Be constantly alert to changing requirements or dramatic new opportunities that could affect the environment in which the developed system is to operate.
5. Perform all the above without taking over the day-to-day management of the project from the SPO or the working managers.

Clearly, not every level of oversight management will perform all five. Item 4 is likely to be done at all levels, whereas item 2 will probably be done only by the level nearest to the SPO/PMO.

Large projects involve large expenditures, large work teams, and long periods of calendar time. Because of personnel turnover, incomplete or ill-understood requirements, and change orders, the definition of what the project is to produce changes over time. Thus, an executive who received a briefing on a project at its inception would quite likely find some significant changes reflected in the system that is finally put into operation. Some of the differences between the anticipated and the actual systems may be due to the executive's inability to understand the technical effort, and some of them may be the result of significant technical changes that occurred as the project evolved. Thus, the oversight manager's first challenge is to understand the status of the project with sufficient depth and insight for his own purposes, but without spending an excessive amount of time obtaining the information he needs.

Big projects inevitably suffer from a form of inbreeding: All of the full-time staff, from the project office to the computer operators, become steeped in the lingo, the functions, and the schedules of the project. Projects develop a momentum of their own and may become an end in themselves, with project personnel losing sight of the fact that a product is to be produced and that users or other functions are to be supported. One way to avoid this problem is to have someone not in the project family read the formal project documentation as it is produced. A knowledgeable user is the ideal reader, but for projects involving new technology, a consultant or other outsider can perform this service. The staff associated with an oversight manager cannot objectively accomplish the necessary reviews. Problems of motivation and attitude aside, the staff person tends to begin to play the role of the oversight manager himself and to forget that his assignment is to review, critique, and advise--not to take action.[1]

Robert Townsend once defined the duties of a Board of Directors very simply: "Judge the chief executive officer, and throw him out when the time comes."[2] In a sense, the primary duty of an oversight manager is equally simple. He must constantly assure that the project as currently defined and scheduled fulfills a necessary mission and is cost-effective. Commercial computer developments that have been started too late or scheduled too leisurely have frequently been forced into early obsolescence by the competition. In the commercial marketplace, timing is everything. If the competitors have already saturated the market with a similar product, the development costs will never be recovered. In the military, the situation is somewhat different. Projects that have completed development are rarely scuttled. There is no "competitive market," and a weapon or system tends to be implemented even though it falls short of intended performance.

---

[1] Oversight managers may surround themselves with staff persons who have had recent relevant project management experience. Unfortunately, these ex-managers sometimes attempt to take over the management of the project, rather than translating their experience into the appropriate advisor/confidant role. This is a most undesirable situation, as it convolutes management authority and reduces the efficiency of the program office.

[2] Robert C. Townsend, *Up the Organization,* Alfred A. Knopf, Inc., New York, 1970.

In the military environment, large, unwieldy software has sometimes been installed even though advances in technology have already made the system obsolete. One intelligence support system was developed whose software required a higher degree of reliability than the hardware technology could support. This mismatch caused frequent system restarts, accompanied by extensive rebuilding of the data base. Since these restarts took hours to complete, the system was seldom available, and the entire project finally had to be scrapped in the initial operational stage, even though many millions of dollars had been spent on it. The project personnel had long been aware of the database reload problems, but objectivity had been lost and the system was implemented without regard to the operational ramifications.

It is the oversight manager's responsibility to sense the unspoken assumption or the overlooked detail and to recognize changing strategic requirements or dramatic new opportunities that could affect the environment in which the system is intended to operate. Being higher in the organization, the oversight manager has a better view of the environment than project personnel. Furthermore, he receives briefings and other information that give him insights into the "big picture" that are not available to lower-level personnel. Consider, for example, a development program for a system that required all mission parameters to be loaded on the ground prior to launch. If the oversight manager receives a technical briefing indicating that ground-to-air data links are approaching operational readiness, he should see to it that the project architects receive the same briefing. This would allow the project to develop software that could eventually accept data-link information for dynamic retargeting. Prompt dissemination of information about the availability of new technological capabilities may prevent major and costly revisions to the software under development.

The oversight managers and their support staffs must accomplish their duties without tampering with the way the project is being managed. If the program office and its line managers are failing to do their assigned tasks, evidence must be carefully gathered, conclusions meticulously drawn, and restaffing arranged. There are a variety of management styles, and the manager should choose a style appropriate to the development environment. Thus, an oversight manager or his staff

may not necessarily approve of the lifestyle, housekeeping, attendance, or punctuality of particular individuals in the workforce, but they must be willing and able to judge them on performance, not on personal characteristics.

III. AXIOMS OF OVERSIGHT MANAGEMENT

1. A PROPER PLAN IS THE KEY TO OVERSIGHT MANAGEMENT

Plans for large projects that involve many people are generally more formal than those for smaller ones. Except in cases where the oversight manager is acting as a super program office, coordinating deliverables from several large projects into one enormous weapon system, the program office and the project manager should probably be allowed to choose the planning methodology that best suits their development activities. However, the oversight manager can and should have something to say about the way the plan is constructed.

A project plan serves two functions: It allows the development effort to proceed in an orderly fashion, and it provides the basis for a series of project management reports. Throughout the life of the project, progress must be tracked against the plan, and the plan must be formally amended whenever it becomes obsolete.

In addition, the planning process must be arranged to satisfy the oversight manager's information requirements. The plan and the cost accounting system must use consistent numbering schemes so that "function delivered" and "cost accumulated" are always compatibly reported. All formal presentations to oversight management and all published documentation must be depicted on the plan, along with the technical tasks, to ensure that the oversight manager's briefings are tracked and prepared in a timely manner. This also allows the staff to informally determine the project's status with respect to the next formal briefing.

The review of the formal documentation is an oversight management responsibility. Independent reviewers must be obtained, and they must be given time to properly perform their reviews. If the formal documents are given visibility in the plan, reviews are more likely to be accomplished on time.

The special information needs of the various levels of oversight management must also be accommodated. Some of the oversight manager's activities are driven by an annual budget schedule. Thus he may wish to be given a combination budget and status briefing before he submits

his annual budget and starts to prepare the background material necessary to defend it.

Very large projects often have significant collateral overtones. The oversight manager must anticipate possible debates about sensitive issues and must see to it that reference materials, progress in sensitive areas, and other properly oriented background data are readily available. For example, if an ongoing development project is likely to cause significant changes in the makeup of the workforce at a military base, and if there is a possibility of job displacements, the project plan should recognize these sensitivities and should include testing for potential personnel retraining. Furthermore, status in this sensitive area should be regularly reported. This would be particularly important if the base were located in a geographical area having high unemployment. The oversight manager should be able to prepare for dealing with unpleasant news or, conversely, he should be among the first to know that anticipated problems were not going to materialize so that he could provide reassurance to those who might feel vulnerable in some way.

Finally, the oversight manager and his staff must be technically competent to identify aspects of the project that are pushing the state of the art. Such aspects must be separately identified and planned, since the whole project hinges on their successful technical accomplishment. Project personnel at all levels may be inclined to indulge in wishful thinking about essential technical accomplishments. Knowing that the path of R&D is seldom smooth, they may dismiss a failure as a "small technical setback." But if oversight management is given an opportunity to review uneven R&D progress in an informed way, they may wish to slow down on associated projects (and conserve money), or they may wish to add resources to a lagging area, or they may even wish to fund a separate parallel effort to see whether a different R&D approach might prove fruitful.

An oversight manager cannot choose an appropriate course of action unless he is realistically aware of the status of all key technical areas that are pressing the state of the art. These mainline R&D areas must be separately isolated in the plan so that they can be properly identified and tracked.

2. MANAGEMENT'S INFORMATION NEEDS DIFFER AT EACH LEVEL

The information needed by oversight management differs in both quality and quantity from that prepared for managers who are associated with the project on a full-time basis. Table 2 illustrates such differences.

The differences stem largely from two factors: First, the oversight manager devotes only a small proportion of his time to a particular project. He is concerned with a wide variety of other projects and issues. Thus, he needs information that will permit him to rapidly reenter the project context and to understand the ramifications of events concerned with the project. He may have just returned from an assignment on an unrelated task force or study committee or he may have been deeply involved in another project within his purview.

The many demands on the oversight manager's time force him to manage by exception. Thus, if he could be assured that a report of "No problems" could be accepted literally, he might not even hold a status meeting. However, since the person reporting "No problems" may not have access to certain facts that the oversight manager has, both parties must sit through briefings, establish project status, and determine what the problems seen by one party mean to the other.

3. SPECIALIZED REPORTING TO OVERSIGHT MANAGEMENT DIVERTS PROJECT PERSONNEL

Many project personnel are rightly impressed when legions of vice presidents, senior government employees, or generals and admirals appear for a status briefing. If the briefings are not included in the project plans, the meeting call may come as a surprise and the project office may be unprepared to make a suitable presentation. Preparing information for presentation, doing a dry run of the briefing, and making the needed corrections may occupy senior project managers for a week or more. With managers unable to attend to the project, it may have to proceed on its own momentum for as long as two weeks (the time required to prepare a briefing, plus the time required to catch up on all the mail and overdue decisions that accumulated while the managers were preoccupied). Thus, not only is valuable effort diverted, but critical project decisions may also languish.

Table 2

INFORMATION DIFFERENCES: PROGRAM OFFICES VS. OVERSIGHT MANAGERS

Information Characteristics for Program Offices	Information Characteristics for Oversight Management
Word Choice and Style	
Any vocabulary, including vernacular, acronyms, and abbreviations of the project.	Full word descriptions, plus acronyms and abbreviations common to the project, in standard English, all to facilitate communication.
Information Quantity	
A balanced view of the status of all project activities, plus additional detail for all pacing activities (items on the PERT chart critical path).	A general overview of the status of groups of activities, plus additional detail on activities with high political visibility, high-risk R&D activities, and routine items that appear on the critical path in consecutive reporting cycles.
Background Information	
Complete detail, plus time-aggregated summaries and historical files that can be referenced for estimating purposes.	Digested facts at a summary level, plus intermediate highlighted/selected areas and projections, together with all related assumptions.
Financial	
Up-to-the-minute detailed records of accrued expenses.	Summarized cash expenses, plus pacing open purchase orders (provided this gross level of detail will support projections of cost).
Status	
Detailed tracking of progress against the plan, with additional detail from those areas that are or appear to be in trouble.	General tracking of progress against the plan, with additional detail from areas that are a latent source of trouble or that are or appear to be in trouble.
In-Depth Detail	
Status reports on every detail of every activity.	Detailed status only on those items that are likely to affect project completion.
Report Context	
An insider view, including common-knowledge calibrations of individuals brought about by frequent direct personal contact with the person or organizational component reporting.	Assumption statements, background information, and other data to help explicitly calibrate the bias (if any) built into the information being reported.

To minimize this problem, one large computer manufacturer elected to cut down on the number of meetings held and to use snapshot reports (Fig. 1) to provide oversight visibility between meetings. The project plan divided activities into phases. At the beginning and end of each phase, meetings were held to take stock of the phase just completed and to present the possibly revised plan for the phase about to begin. Following the two back-to-back meetings, a weekly snapshot report would be sent out by the project manager. Upon reading this report, a senior manager who required further information could either discuss the matter on the phone or call for a full briefing that would include all relevant individuals.

A snapshot report has several interesting properties. First, its content is derived from the list of key deliverable items being produced in the current phase of the plan. Second, it is a fill-in-the-blanks report, requiring project managers to spend no more than a few minutes a week filling in the blanks. The required number of copies are then routinely prepared and distributed. Third, after the recipients have seen a few of these reports, they can ignore any narrative and read what is filled in the blanks. The report format becomes a form of shorthand that a busy executive can comprehend without detailed study.

Overzealous staff personnel frequently insist that project management prepare briefings in some arbitrary format that is presumed to be ideal for an oversight manager. While such stylized briefings may present information in a style to which the busy executive has become accustomed, they frequently cause serious problems: (1) They may drain valuable and often scarce effort from the project; (2) they may cause the project staff to gather, digest, and summarize information solely for the executive briefing; (3) the content of such briefings may fail to reflect the current position in the project plan. There may be some standing agenda items to be reported on regularly, but many of the interesting topics that should be discussed are related to the state of the development and what is being currently undertaken, e.g., statistics

Software Study Project
Bldg 13, M.S. 05
Ext. 479

6/6/73

MEMORANDUM TO: Project Steering Committee

Subject: SOFTWARE MAINTENANCE STUDY GROUP
Weekly Status Report No. 5

From: S. J. Morehouse

cc: J. L. Madden, W. H. Chambers

Please excuse this format; to get you status information each Friday without excess work, I have filled in the blanks on this standard message.

1. The Study Group consists of 18 members plus secretary.
2. During this week we have interviewed: 6 branch offices, 4 change teams, 2 support centers, 3 computer-based support systems. Re-interviews (are) (are not) a problem.
3. The charting and process documentation is (keeping up with the interview) (running late) (losing ground).
4. There are 37 ideas/suggestions in the file, up from 33 last week.
5. There are 63 documents in the Library, up from 59 last week. The backlog of unreviewed documents stands at 21.
6. Firm interview schedules are in hand for 3 weeks in the future. We (are) (are not) having problems with cooperation, support, or study personnel.
7. No key memos/reports are attached to this status report.
8. Highlights/problems from this week are:

ROUTINE WEEK, NO PROBLEMS.

Fig. 1 — Actual snapshot report

on the documentation are meaningless during the programming phase but extremely meaningful during the testing phases close to initial operational capability.

4. DATA FOR OVERSIGHT MANAGEMENT MUST HAVE ASSURED INTEGRITY

Every experienced database manager knows that data that are not used are effectively undependable; it seems to be a form of Murphy's Law. Regardless of the systems and procedures, the training, and the input checking and data verification, data placed in an organized file to be used at some uncertain future time are never ready when they are to be referenced. Errors inevitably creep into the database and render it useless.

The situation is similar in the case of management reports. If line managers use the data to manage the project, the data can be aggregated and accepted with confidence by higher levels of management. A supervisor who finds the status of a programming team erroneously reported will insist on a correction, because it affects his career and his professional standing. Unless such feedback takes place, no one will be aware of what is being said about individual parts of the project or what errors (typos, computational mistakes, etc.) may have occurred. Mistakes in aggregation can go unnoticed and can in turn mislead executive management and cause them to lose confidence in the project. In this example, the importance of the data to the project and the built-in mechanisms to verify accuracy and completeness jointly assure the integrity of the data for oversight use.

If executive management insists on its own specialized reporting, project managers will have to give increased attention to the information reported up the management chain. Since such managers already devote 100 percent of their intellectual energy to the project, special management reporting requirements dilute the technical effort that can be applied to project details.

There are two alternatives to this dilemma. First, executive management can be trained to use the data derived from statistical and financial reporting and from status information routinely circulated

within the project. Second, if oversight management has some unique reporting requirements, the project staff can modify or adjust its procedures to accommodate such needs. Internal project reporting systems can usually be rearranged to produce the information oversight management requires for special visibility.

As a general principle, data can be considered reliable for oversight management only if they are important and are used by the project supervision, or if special attention is given to assuring integrity.

Against the background of these four axioms, a series of case histories are presented in the next section, to develop principles that are directly relevant to Air Force oversight management.

IV. PRINCIPLES FROM CASE HISTORIES

This section draws on experience to illustrate some important lessons for software management. The case histories described below are of necessity anonymous treatments of real industrial projects. In each, the details of the mechanisms used for oversight of the project are accurate. Following the characterization of each case study, the relevance of the results to Air Force applications is summarized.

CASE 1

Discussion

It is an accepted fact of life in computer system development that a computer program of any size is never fully checked out. This situation has existed for over 30 years, and despite attempts to design error-free software and to produce thoroughly tested software, the complexity and size of the programs needed in operational systems still exceed the ability to produce them perfectly. Thus whenever a large computer program is delivered, the builder must provide a support organization to investigate error symptoms as they occur, to identify the problems, and to solve them. Software for the civilian market is analogous to that for the command-and-control community in that both are complex and large, and neither can be exhaustively tested, because the number of test cases would be too large. In addition, the configuration of hardware, software, and personnel at the development facility is often different from that at the operational site(s).

By the time a manufacturer has installed a few thousand copies of his operating system, he needs a formidable support organization to investigate symptoms, isolate bugs, make repairs, distribute changes, and install updates. At this stage, one large computer firm decided to improve its software maintenance process.[1] Of itself, this is a major software and system development project.

---

[1] "Software maintenance" is the common term for ongoing support. The Air Force now uses the alternate and more appropriate phrase "life cycle support."

A one-year study defined the existing system, the alternatives for improvement, costs, and time schedules. A two-year project effort then implemented the 20 or so unique subprojects that were subsequently installed in the deployed worldwide installations. The project team consisted of about 50 persons who worked, 25 at a time, on the two principal phases of the study. A project planning and status system was essential to keep 25 senior and experienced personnel working on a tight time schedule and to be sure the wishes of three senior corporate executives who shared responsibility for software support were understood and reflected in the project's progress.

The project plan was based principally on a simplified form of the formal PERT chart, known as *dependency charting*. It was graphic rather than tabular and was based on an exhaustive enumeration of the "tasks to be accomplished." The tasks and their primary dependencies were arrayed on a single piece of paper with a linear calendar on the bottom, and the chart was formally updated whenever any change in the tasks was required.

Although the principal planning tool was the centralized dependency chart, each task on the chart was accompanied by a page in the project notebook which described in detail the work called for under the task. Specifically included were the

- Task name: a one-paragraph description of the task at hand.
- Predecessor tasks: those whose completion is required before the present task can itself be completed.
- Successor tasks: those that are in turn dependent upon the task at hand being completed.
- The skills required for the successful completion of the task.
- Any special facilities required for the accomplishment of the task.
- The number of calendar days required to accomplish the task.
- The target start date for the task.

A medium-sized project will include several hundred tasks, and the dependency chart may be 3 feet wide and 30 feet long. Although the project manager requires such detail, the oversight manager merely needs to be assured that the planning is done to the required level of detail and that all status reports are illustrated, using the large chart as a backdrop.

A typical oversight management meeting might be held in a room where the chart can be posted on the wall, with several copies of the project books containing task descriptions available for reference. If the project manager has posted the accomplishments of the reporting period on the chart (see Fig. 2), he will be able to indicate any tasks that are lagging or leading the current time line.

It is very unusual for a large project to be conducted exactly as initially planned. The charting/notebook technique allows the oversight manager to understand the changes proposed to the work plan and the impacts of these changes on the resources required.

In the software maintenance improvement project cited above, just such a planning technique was used. When the three corporate vice presidents met in plenary session, the chart posted on the wall had the completed tasks colored in yellow. The lines depicting the completed tasks moved from left to right as time and accomplishment progressed, and lagging tasks were instantly apparent. The oversight manager could consult a reference copy of the task book for further definition of the work to be accomplished and could see the downstream tasks that were likely to be impacted if the bottleneck were not corrected. For example, if a computer facility required for testing were delayed, the extent of the delay and the need for overtime, priority, or other special consideration could be discussed. On the other hand, problems related to staffing, politics, approvals, or technical activities could be discussed, focusing on productive issues.

Each executive who came to the meeting was, of course, concerned about the performance of the activities under his direct responsibility. Each one had to determine whether any of the lagging tasks were either his fault or likely to cause him to change his planned course of action. Each oversight manager was also concerned about slippage of any major

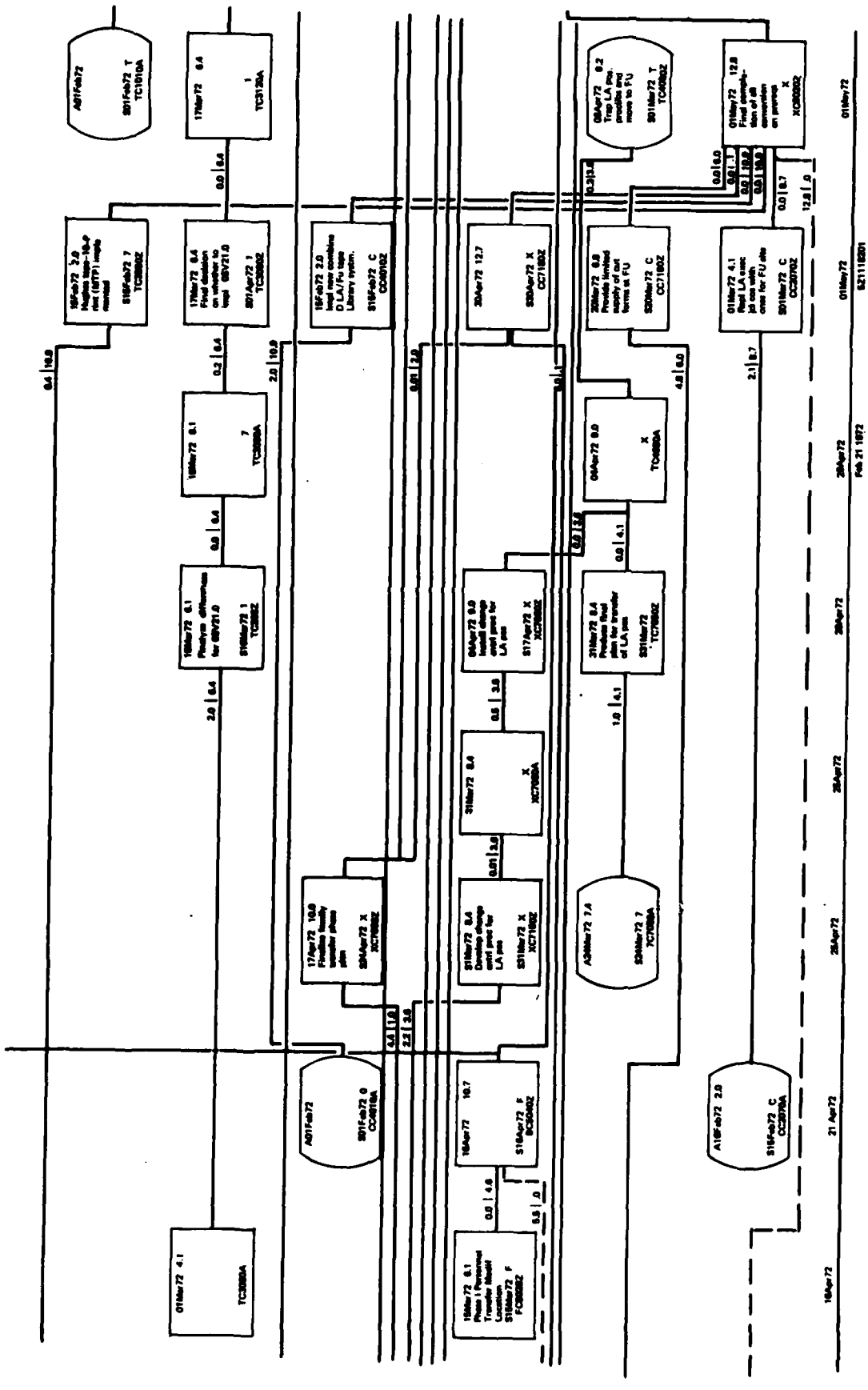


Fig. 2 -- Section of a 13-foot dependency chart (full chart shows the 164 tasks involved in moving a computer center)

milestones that might affect his political situation. With both the plan and the status graphically presented, each oversight manager could privately review the project's status with respect to his own personal viewpoints.

The attractiveness of the approach was instantly appreciated. The executives took no more than 10 minutes to grasp the meaning of the plan; and in subsequent sessions, the meaning continued to be readily apparent without further retraining or reeducation.

Relevance to Air Force Applications

There are two major criteria for effective oversight management tools:

- *The meaning of the information must be easy to assimilate upon initial introduction.*
- *The relearning time during subsequent presentations must be minimal.*

The same planning technique has been used in many other industrial situations. Two vertical time lines on a dependency chart define a project phase precisely; the tasks lying between the two lines are those to be accomplished within the defined phase. Before an oversight management meeting, an administrator can review the tasks and enumerate those that produce deliverable items. Thus the oversight manager can review, at any level of detail he chooses, the work to be accomplished during a phase and the visible accomplishments from a series of tasks.

CASE 2

Discussion

On one large contract, phases were declared complete as the money allocated to each was used up. Thus, incomplete tasks cascaded from phase to phase through calendar time. Although the project was superficially on schedule--phases were apparently completed, as measured by time and budget--a readily foreseeable disaster was going to occur, and in fact, did.

Many large military contracts have been awarded with project phases incompletely defined. If the contractor's payment schedule is related to phase completion, such contracts may encounter trouble. Project office personnel may use phase definitions plus the accompanying progress payment as a kind of incentive to motivate the contractor to accomplish the work in some other order than was initially planned. If progress monies are held back until a phase is completed, the contractor will always be interested in having a phase declared complete. If a project exists in a highly charged environment and is lagging for one reason or another, both the SPO and the contractor are extremely interested in having phases declared complete. Oversight management review can insist that a project's status be truthfully and realistically presented.

The oversight manager is the check and balance on any tradeoffs made by the SPO and the contractor. Project plans do change, and this should be expected. However, until a plan is formally amended, the oversight manager should insist that the project be conducted in accordance with the plan, that all of the deliverables be produced in the phase defined by the plan, and that the phase definitions be held as established.

Relevance to Air Force Applications

This case clearly demonstrates one of the major principles of oversight management:

- *The overall responsibility of oversight management is to promote integrity and forthrightness in the conduct of a project.*

CASE 3

Discussion

The chief executive of a large, reputable financial institution concluded that the evolution of his business had turned it into an information company. While he still ran an international banking

organization, still managed several billion dollars worth of investments, and still dealt with about 500,000 customer transactions each day, he believed his business was more an information company dealing in financial information than a traditional banking enterprise. There was adequate evidence for such a conclusion. He had an international communications network for funds transfer, letters of credit, and other negotiable paper; he had a separate network that provided information on the markets of the world; and he had yet another network that handled interbank clearings on behalf of individual customers. With about 50 computer centers and about 10,000 trained computer personnel, the investment in computers and communications facilities rivaled his investment in buildings.

The chief executive officer foresaw changes in American banking regulations that would require him to become more competitive if his business was to survive. However, his firm was inhibited by a lack of computer literacy on the part of most of its executive managers. While computers and communications were becoming increasingly important to the business, the executive management (the division president and vice presidents) were losing ground in terms of their ability to oversee affairs.

The company also had symptoms of more immediate troubles. While the turnover in the line computer organizations was not unusual, the turnover among senior computer personnel was excessive. The senior computer people reported directly to the executives, who lacked computer fluency, and great frustrations built up. There followed friction, personnel reassignments, and significant attrition as senior people terminated for more satisfying positions elsewhere. As might be expected, large development projects suffered as management changed and as the incoming management redirected the effort. One large development project was initiated for the third time. The inside view was that its first two false starts had wasted more than $10 million.

When a new chief executive with a good understanding of computer development principles was installed, things improved. An executive

steering committee was set up to review all projects exceeding $10 million in development cost or affecting more than two operating units. A computer staff organization, over the chief executive's signature, established a standing agenda for steering committee meetings. Each project manager had to conduct his own meetings, and technical reports were to be given by the persons accomplishing the work. Each report had to include (a) accomplishments since the last meeting and related expenses, (b) any problems that were visible to the project staff, (c) highlights of what was to be accomplished before the next meeting, and (d) an independent assessment of the project's targets against those of the competition.

Since it cannot be assumed that senior executives have total recall, 10 days prior to each steering committee meeting a background pamphlet was provided to each oversight manager, refreshing his memory as to the project's goals, size, current phase of endeavor, completion date, historical financial status, and cost projections. Such background booklets were usually 20 to 25 pages in length and allowed a manager to renew his thinking on the project in a few moments.

The first several meetings generated considerable unhappiness and complaints. They were seen as an intrusion into the purview of each responsible executive; the background booklets added to the busy executive's priority-reading backlog; and the meetings themselves intruded on everybody's already crowded calendars. However, the chief executive insisted on this arrangement, and each executive eventually found that the process was democratic, since all development projects eventually came under scrutiny. In addition, each one learned from the mistakes of the others and some of the development lessons turned out to be transferable. In effect, the procedure was an unstructured on-the-job course in computer system oversight. Finally, as the senior executives became more aware of pertinent computer project issues, each was able to ask better questions of his own personnel. Over time, fewer surprises were uncovered during the steering committee briefings.

In contrast, a large aerospace corporation held only expenditure approval meetings. The chief operating officer met with his senior

staff--chief engineer, controller, head of manufacturing, etc.--on a regular monthly schedule to approve system development proposals. They went through the motions of reviewing such proposals and approving them for initiation. However, the operating officer controlled the agenda and in preparing it gained quite a bit of background lacked by the other executives. Since he was the most computer literate of the group, he naturally dominated the proceedings--he was the senior manager, he was the most knowledgeable, and he had done his homework. Every other member was hearing the proposal for the first time. Needless to say, the meetings were rubber-stamp actions; the only real thing accomplished was the certification by the controller that the desired funds were or were not available on the time scale requested.

Relevance to Air Force Applications

This leads to an additional oversight management principle:

- *Information provided to oversight managers must be in harmony with their backgrounds, experience, willingness to contribute, and ability to understand.*

CASE 4

Discussion

Management at a high-technology electronics firm believed that major computer expenses should be treated just like any other expense. Thus the same review and approval mechanisms used for building programs, new plant startup, and new business were used for computing expenses. While the established procedure has a certain superficial appeal, it assumes an unrealistically high level of maturity computer affairs. Seldom does a building fail to perform and hence become useless; but ill-conceived and ill-managed computer developments frequently conclude that way. Buildings and facilities are constrained by federal, state, and local regulations; thus the real estate and construction departments of major corporations are stable and have long-standing principles and procedures to guide them. In contrast, many multiyear, multimillion-dollar computer developments have been started by a small cadre that has

to hire an extensive staff or subcontract more than 90 percent of the work to be accomplished. The effort, therefore, cannot benefit from programming standards and long-standing in-house project management methods.

Relevance to Air Force Applications

An executive management that treats computer development projects as one-time financial expenditures denies itself a significant learning experience and runs the risk of having well-intentioned employees launch vast projects with inadequate preparation. Thus, another principle of oversight management:

- *The review function provided by oversight management should continue throughout the lifetime of a project, although the periodicity of meetings can and should change to match the pace, and possibly the stage, of the endeavor.*

As one chief executive observed, "Decisions critical to the business enterprise are too important to be left solely in the hands of technicians."

V. A MENU OF PROVEN OVERSIGHT MANAGEMENT TECHNIQUES

The function of oversight management is only now being recognized as a separate management discipline pertinent to the system development process. Therefore there is no established body of knowledge about the needs of oversight managers. The following techniques illustrate arrangements that have been made and suggest general approaches that are applicable.

All of the techniques discussed here meet the following criteria:

- The technique must be in regular use by line management and must provide useful insights to the oversight manager.
- Information must be derived from data readily available to line management.
- The benefits to oversight management from available information must greatly exceed the cost of preparation.
- The techniques must have been used in one or more real situations, and the benefits must have accrued as planned.

No relative importance is implied by the order of presentation. Many of these approaches have, in fact, been successfully applied in combination.

1. **Snapshot reports.** A "snapshot report" encapsulates project status information for the busy oversight manager (see Fig. 1). It is easy to prepare, easy to assimilate, and very informative.

2. **Dependency charts.** These charts serve as a useful tool for project planning, status reporting, and project replanning (see Fig. 2). While they require time and effort to produce, the resources involved are not significantly greater than those for any other planning method that involves similar detail and similar insight on the part of the project leader. The principal advantage of dependency charts is that their simple notation and graphic form allow both the project office and various levels of oversight management to rapidly appraise the project status and relate it to opportunities, delays, or necessary redirections.

Tabular forms, which are popular for project planning and status reports, can provide almost as much information to the project manager as graphic presentations, but lengthy computer printouts are difficult for an oversight manager to assimilate. While many of the tabular forms are supported by computer programs which aid their creation and ease their maintenance, a manager must have considerable training and experience before he is able to glance through yards of computer listings and perceive the full impact of a late task. Ideally, the project manager's planning and status tracking computer program would produce a dependency chart in graphic form, and would produce a new one after each update of the status file.[1] While the graphic approach may be optional for line managers who spend full time on the project, it is essential for the oversight manager.

3. Dynamic skill-mix adjustment. Some system development projects require a fixed mix of professional skills throughout the life of the project. In these cases, the project should be staffed properly at the beginning, with additional people being added subsequently to compensate for attrition. In other cases, the skill mix varies significantly from one phase of the project to the next. Designers, system programmers, hardware specialists, data administrators, crypto-programmers, etc., may be required to work for a while, and then be reassigned. When a project is planned, a given skill mix is specified; but projects have often developed trouble when the planned or actual supply of people or their distribution of skills does not match the dynamic needs of the project as it proceeds.

If a dependency chart has been prepared with a linear calendar as its time base, summaries of required manpower by skill type can also be plotted on the same time scale. At each project status meeting, the oversight managers have information about planned changes in skill mix. If current staffing is reported by skill type, improper staffing--a frequent cause of project slippage--can be circumvented.

---

[1] One software package that has this ability is EZPERT from Systonetics, Inc., Fullerton, California.

4. Project workbooks. In *Mythical Man-Month,* Frederick Brooks described the project workbook that served him well in the development of the OS/360--an effort that involved 1,000 people, pushed the state of the art, and had a tight time schedule.[2] Since the mid-1960s, project workbooks have been created by many other projects in many other environments. They have always provided the benefits claimed. A project workbook can be thought of as a special kind of filing system which serves the development project and provides reference materials for the subsequent maintainers of the system.

A successful workbook must be based on a table of contents prepared by capable individuals before the project starts. It must exhaustively enumerate the types of documentation the project will produce; it must enumerate the ways project personnel will need to access the body of knowledge. A workbook is primarily an aid to communication. However, if a technical problem comes to the attention of an oversight manager, the workbook provides an instant source of reference material. To the technical auditor, a workbook is invaluable.

5. Independent reviews. An objective reviewer should read all external documentation the project produces. This individual must have the background required to understand the material. Ideally, the reviewer should be drawn from the end-user organization that will eventually operate and use the completed system.

Document reviews are really in the nature of an elephant hunt; the search is for gross oversights. While critiquing writing style is generally inappropriate, an excessive amount of minor inaccuracies, oversights, omissions, etc., is a sure signal of substandard quality. The experienced reviewer must know when to sound the alarm because of an excess of minor problems as well as when to sound it because a specification does not have performance or response time targets, or an operational manual fails to discuss all of the conditions of restart, or training materials are inappropriate for the intended audience.

---

[2] Frederick P. Brooks, Jr., *Mythical Man-Month*, Addison-Wesley Publishing Company, New York, 1975.

6. Comparison with similar projects. Frequently, oversight management can profitably compare the project under consideration with "similar" projects done elsewhere. For a project that is larger or more complex than earlier projects of the same type, or that presses the state of the art, or that has more political ramifications, a small study team should be given a few months to prepare a meaningful comparison report. The difficulty comes in resolving the definition of "similar." Projects have many external attributes--size, cost, time schedule, relevant experience of the technical leadership, numbers of vendors involved, numbers of user organizations involved, size of the database, size of the communication network, numbers of development items being reduced to common practice for the first time, etc. Given a fairly exhaustive list of attributes, the study team should choose a list that discriminates between the project under consideration and any other being compared with it. The list of attributes for comparison, or the *characteristic vector*, can then be quantified on appropriate scales.

The study team can identify projects that are similar to the one under consideration, and a characteristic vector can be prepared for each. When projects are identified whose characteristic vectors indicate meaningful similarity with the project being studied, contacts may be needed with personnel from the comparison projects to obtain data to complete the vector; published material may be inadequate. For projects whose vectors compare favorably, such attributes as costs, schedule, and staffing can be reviewed to determine if the proposed project plan is reasonable.

Some years ago, proponents and opponents took diametrically opposite views of the proposed Air Force Advanced Logistics System (ALS). Some thought it was unique in every way, while others were convinced that it was a routine development that just happened to be large. The argument was resolved by building a characteristic vector for ALS and for similar projects. The characteristic vector clearly identified the aspects of ALS that were routine and those that advanced the state of the art, and hence required special management attention.

7. Project glossaries. Every project of significant size has training courses for new people and a publications team to produce external documents. The manager of any project that is large enough to support both of these functions should create a project glossary. The glossary supplements the normal idiom of the application field (command and control, logistics inventory control, tactical air support, etc.), so that new people can read project documentation with understanding and so that the project documents produced will use a consistent vocabulary.

A project glossary that is kept current is extremely useful to oversight managers. It is an essential reference document for the oversight manager, because project memos and documents are inevitably written in the vernacular peculiar to the project.

8. Process charts. If the project goal is to upgrade an existing system--particularly one that is large, complex, and ill-documented, or has multiple interfaces with other systems, or provides continuity of operational support--a set of process charts that describe the behavior of the existing system in complete detail can provide benefits that far outweigh the costs of producing them.

Process charts originated in the automotive industry, where they are used to show all the materials flow and manufacturing actions necessary to complete an automobile. In the context of a computer system, process charts would show the flow of data both into and out of the system, the computational processes that occur within the computer, together with all their data inputs and outputs, the actions that are taken by people, and the interrelationships among all of these factors. A process chart usually has a horizontal linear time scale at its base; parallel paths of individual process steps are drawn from left to right across the chart. In the automotive industry, entries at the left edge of the chart include raw materials, product information, and purchased components. At the right edge, products such as finished automobiles, maintenance manuals, and replacement parts appear. In the computer-system context, entries at the left edge would include data in the form of keyboard actions, magnetic tapes, or punched cards, operator or system personnel actions, and schedules. Deliverables at the right edge might

include printed paychecks, tabular printouts, magnetic tapes for other systems, graphic displays, or printed documentation. All the process actions and other steps necessary to transform the inputs into deliverables are depicted between the left and right boundaries. A software process chart might show only the names of software modules in the sequence of execution; a more detailed chart could show the flow of data through each module. A process chart may be annotated with organizational responsibility, cost, or other indications of effort required to complete a single process cycle.

Using a process chart, an astute manager can rapidly become familiar with the existing system. The chart can also serve as a reference document defining the processes required to produce each deliverable item. Finally, it is a useful baseline document when process changes are being considered; a chart depicting the current process sequence can be contrasted with one depicting the modified process. For example, consider a support system for intelligence analysts that is batch-oriented, must support an ongoing operational responsibility, receives inputs via punched cards, and has been reporting to other systems via magnetic tape. It is to be converted to an on-line system that will provide the analyst with semiautomated interactive support tools, will receive input electrically, and will deliver information electrically to other systems as well as permit limited interactive inquiry from other systems. If the original system is not well-documented, process charts portraying all events in that system must be created before the follow-on system can be designed. These charts can also be a valuable asset for the oversight manager, to assure that essential attributes of the initial system will not be lost in the new one, or that the transition from one to the other will be smooth and without interruption of military support.

Statistical reports or production and financial reports are often not well coordinated. Costs--by deliverable event--are not time-synchronized and possibly not even related by a common time scale. A process chart can be used by the comptroller's office to check its accounts and to verify that the expense codings on them correspond to the sequence of processes leading toward products. This provides direct cost tracking that matches the natural way progress is achieved.

9. Skill wheels. The skill wheel (see Fig. 3) was conceived to graphically represent any key job and to allow the oversight manager to rapidly determine whether a candidate or an incumbent is qualified to fill it. For example, in early reviews, the oversight manager may need to calibrate the skills of key personnel against their jobs to assure himself that the effort is in good hands.

After a traditional position description is prepared for any key position, the job functions can be annotated, and the proportion of time in a normal month each function should require can be estimated. Thus, if a job is primarily administrative and has few technical activities, the total of the administrative percentages will exceed the total for the technical activities.

A skill wheel is constructed in the following manner: A pie-shaped segment is shown for each principal job function. The angle included in each segment is the activity percentage translated into deg . Thus, if a job were 50 percent technical and 25 percent administrative, 180 degrees of the skill wheel would be technical and 90 degrees would be administrative. The typical wheel for a key project position frequently has seven to twelve facets; the multifaceted nature of the job makes it difficult to fill.

After a skill wheel is constructed, the resume of an incumbent or a job candidate can be mapped onto the wheel. A person with a very strong technical background in a technical job might have the entire technical fraction of the skill wheel completely shaded. If the same person had no administrative or management experience, the shading in these areas would be light or nonexistent.

When reorganizing, skill wheels can also be used to avoid the possibility of structuring a job in such a way that it is impossible to fill. If the technology or the political environment demands that a job be structured in a certain way and this keeps it from being filled with a single individual, the skill wheel allows management to define the person whose skills are required to complement the incumbent.

Oversight management must be constantly aware of the risk of unrecognized skill mismatches in key jobs. When this occurs, the person who is assigned to a job he cannot fulfill may resort to a strategem

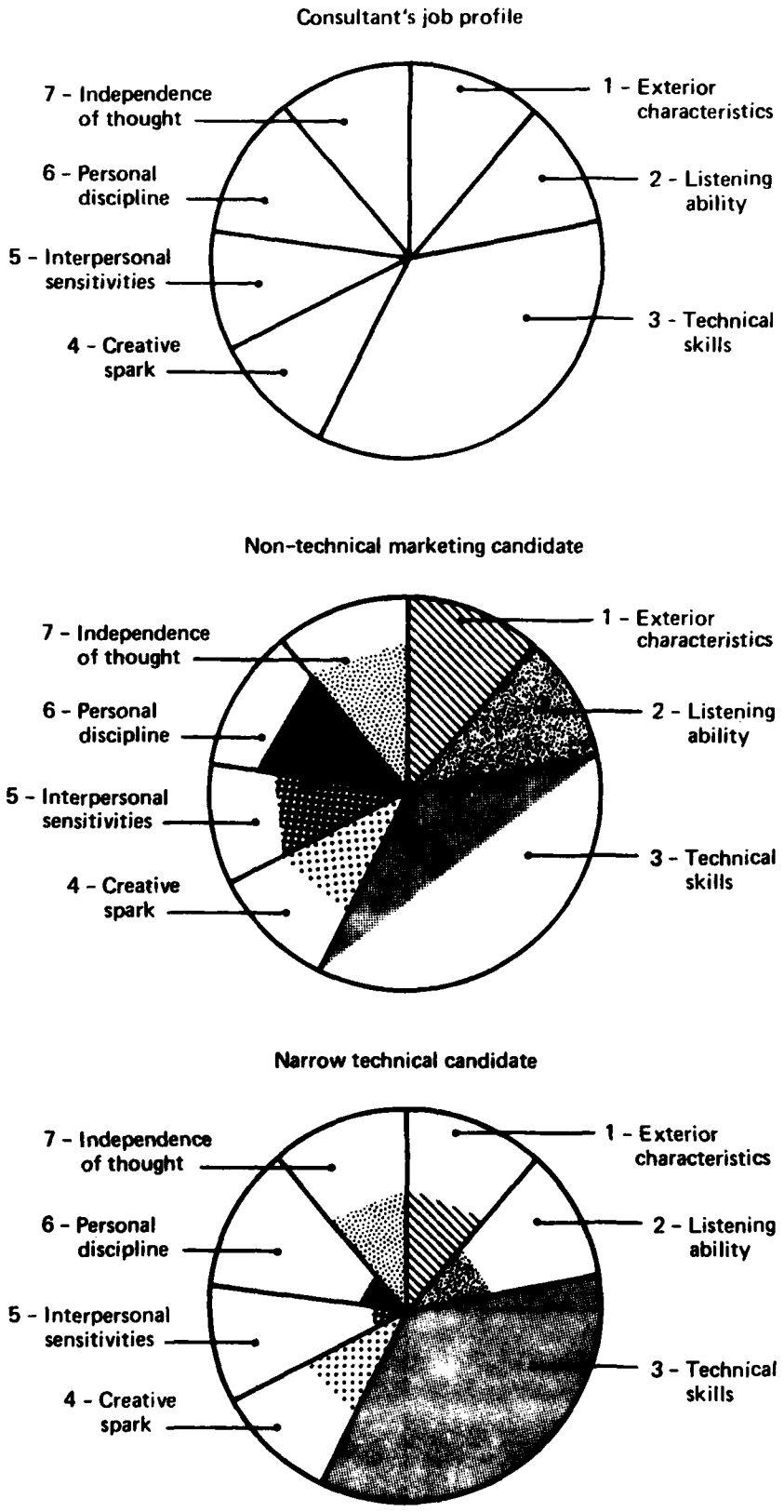


Fig. 3 — Skill wheel used to depict a consulting job and two unqualified candidates

that will be catastrophic for the project. He may become bureaucratic to cover his lack of knowledge; he may bury himself in technical activities to avoid feeling guilty about not performing the rest of the job; or he may recognize his failure and prepare his resume--and finally leave before his secret is discovered.

10. Economic modeling. The oversight manager should be able to obtain reliable answers to questions such as the following:

- If I authorize the requested overtime to make up for a schedule delay in one work unit, what will be the impact on project completion time and cost?
- If I authorize additional computer support (terminals, personal computers, or work stations) and if the project delivers the additional quality and quantity as proposed, will the extra expense be exceeded by reduced cost, shortened schedule, or increased quality (reduced maintenance expense)?
- At the rate we are addressing requirements, producing modules, and/or testing modules, is the schedule for this phase still reasonable?
- Given the proposed schedule revision and the proposed staffing, what is the expected impact on project cost?
- Using available historical data on software maintenance (errors found in testing vs. errors found in service) and given early test experience on the new system, are field maintenance activities still within expectations?

On very large or very complicated projects, oversight management is frequently confronted with the need to choose among several alternatives, no one of which is an obviously superior choice. The tradeoffs related to these decisions can sometimes be clarified by building an economic model.

Modeling can be done at many levels of detail and complexity. Simple paper and pencil models are sometimes adequate to determine tradeoffs between two courses of action. Models that can be constructed on a personal computer are usually adequate for all but very large

projects. Software is available to aid in model construction, and simple spreadsheet calculations may suffice for highly aggregated analyses.

The function modeled can be either cost or time or both. If a chart has been made of the development process, the coefficients for either a cost or time model may be readily available. In other cases, an analyst may have to work a week or two to prepare the basic model.

11. Requirements tracking system. If a project has a structured requirement specification, and if there is a requirements tracking system[3] linking every feature of the design to the requirement that generated it, the total number of requirements and the number covered by one or more design statements can be used to discern trends that allow the oversight manager to track the progress and thoroughness of the design process itself.

For example, if a requirements document has identified 1,000 functionally required features, and if the design progress reports on consecutive months address 300, 500, and 650 of these requirements cumulatively, the design is probably progressing satisfactorily. However, if late in the design, the number of requirements not addressed becomes constant for any significant period of time, this may indicate that the designers are having trouble meeting the specifications. Similarly, if the number of requirements to be met tends to increase over time and fails to stabilize, the designers may be asking questions about items that were omitted from the original specification. Errors or omissions from the requirements document could lead to major project upheavals.

12. Rate charting. During the programming phase, useful statistics on the programming process can be produced with a technique known as rate charting, which was developed simultaneously some years ago at a large newspaper and at a large aerospace firm.[4] By maintaining time plots of the number of named program modules, the number of modules coded, the number of modules that have had one test run against them,

---

[3] Robert A. Pierce, "A Requirements Tracking Tool," *ACM Software Information Notes*, Vol. 3, No. 5, November 1978.

[4] Terry R. Snyder, "Rate Charting," *Datamation*, November 1976, p. 44.

and the number of modules that have completed unit test and are ready for integration (all explicitly determinable events, provided the development support system has anticipated the needs of oversight management), meaningful time trends, such as growth of test cases versus expected number, or growth of requirements versus the initial specification, can be shown.

In one project, the number of named modules continued to grow as the programming process progressed--a clear indication that the design was in trouble. When a technical audit was performed, the project was canceled. In another case, the number of named modules increased slightly but became stable, after which the number of programs completing various development milestones continued to increase and eventually equaled the number of named modules; this project was found to be progressing satisfactorily.

With rate charting (either manual or automated), the status of projects containing several thousand modules can be quickly and effectively monitored by an oversight manager.

13. Test case library. The testing process is extremely critical during system development and frequently accounts for as much as 50 percent of the development expenses. One manufacturer has prepared a test support system that contains a growing library of test cases produced by the test director and his associates. It also contains a growing library of program modules that have passed through unit test and have been certified by the appropriate development manager. As the test team applies tests to modules, statistics are produced which can be useful for oversight management, providing answers to questions such as

- Is the test library growing or stable (an indication of the quality of test development)?
- Has every test case been executed at least once?
- Has every module been subjected to at least one independently developed test case?
- How many modules/test-case combinations have been satisfactorily completed?

- How many modules/test cases failed to run to a natural conclusion?

Such questions help relate actual status to that anticipated during initial project planning.

14. Progress statistics. There is a computer program called Optimizer-III,[5] which accepts a COBOL program under test and inserts counters at every branch point. When a program that has been so modified is tested, the counters record which paths are exercised by each test situation. This is a useful tool routinely used by commercial programming development organizations. If the statistics produced by Optimizer-III are summarized, they provide useful information allowing management to compare actual and expected status. For instance:

- How many of the modules that have been released for testing have started the testing process?
- How many of those modules in test have successfully executed sufficient test cases to exercise each code path at least once?
- How many of the "thoroughly tested" modules still have a part that has not been executed?

If such progress statistics are related to a machine accounting system that collects costs, useful planning parameters can be produced, such as elapsed time for testing a thousand lines of code, or machine time required for testing a thousand lines of code. If more sophistication is required, such information can be plotted against the number of branches in a module, which is one measure of complexity. Such parametric measures (each requiring careful interpretation) throw some light on whether the project's basic schedule is optimistic, pessimistic, or about right.

Care is essential in interpreting statistics derived in this way. Each statistic must be accompanied by a series of assumptions, caveats, and background information. However, oversight management has

[5] Capex Corporation, Phoenix, Arizona.

traditionally relied on similar statistics (e.g., cost per pound of combat aircraft during World War II, or classical manufacturing learning curves) to provide gross insights that are not attainable any other way.

15. Oversight manager's workbook. If the substantive materials and techniques described above are collected and organized into an "oversight manager's workbook" (a direct analog of the project workbook mentioned earlier), a project history will result. A central file of such histories will allow the extraction of planning factors that are useful during early discussions of new development projects.

One industrial construction firm has maintained project histories for years. Given the physical parameters of a building site, this company can estimate cost within 10 percent without ever seeing the plans or visiting the location, using a historical record of several similar factories. The estimate can be further refined by literally flying over the proposed site. To get a fixed price bid within 3 percent, they need to review the plans and calculate costs accurately. If historical project histories work for the construction of breweries, mayonnaise factories, and corrugated-paper plants, they should be correspondingly useful for providing cost baselines for computer development projects.

* * * * * *

These fifteen techniques, which admittedly are not a comprehensive inventory of all that have been used or that might be conceived, can be used singly or in combination for various activities in which oversight management might find itself involved. Figure 4 suggests useful opportunities for applying them to the oversight actions noted at the end of Sec. I.

	Snapshot reports	Dependency charts	Dynamic skill-mix adjustment	Project workbooks	Independent reviews	Comparison with similar projects	Project glossaries	Process charts	Skill wheels	Economic modeling	Requirements tracking system	Rate charting	Test case library	Progress statistics	Oversight manager's workbook
Initial approval		•			•	•		•		•					•
Contractor selection			•						•						•
Key people selection			•						•						•
Project estimation		•				•									•
Monitoring schedule	•	•									•	•	•		
Monitoring funding		•													
Technical details				•			•	•							
Capabilities				•	•		•	•		•					
Skill matches			•						•						
Schedule and funding										•					
Quality vs. requirements				•	•						•				
Phase monitoring	•	•					•					•	•	•	
Project reorganization		•				•			•						
Project redirection		•		•		•	•	•		•					•

Fig. 4—Oversight technique applicability

VI. R&D POSSIBILITIES

We believe that there are important steps that AFSC and the Air Force can take to improve the effectiveness and thoroughness of the many reviews of software-intensive projects. We believe also that the capability of the Air Force oversight process can be much better tuned by using the techniques described here for perceiving latent problems and ascertaining the true status of a project.

It is a truism that computer projects, whether they are embedded in large systems or are simply large or visible in themselves, require periodic executive review by oversight management. The information needs of the oversight manager are different from those of project managers, even though some of the oversight manager's data can be derived from those routinely used by project management. We have identified the following principles concerning the oversight manager and his information needs; there may well be others.

- Information provided to oversight managers must be easy to assimulate upon initial introduction; the relearning time required during subsequent presentations should be minimal.
- Oversight management must promote integrity and forthrightness in the conduct of the project.
- Information provided to oversight managers must be organized and presented in a way that is appropriate for the managers' background, experience, willingness to contribute, and ability to comprehend.
- Examinations by oversight management must continue throughout the entire life of the project.

These principles lead to some essential properties of management tools intended for use by oversight management:

- Tools should use data extracted, wherever possible, from the detailed information the project management routinely uses.
- Any oversight management feature of a tool should be optional for project management, which should not have to produce, review, file, etc., such information for its own use unless it chooses to do so.
- Where the above criteria cannot be met, information especially prepared for oversight management must be accompanied by integrity measures to assure accuracy and completeness, and all assumptions underlying information elements or aggregates must be stated.
- Specially prepared information must provide substantial insight relative to its cost; otherwise the drain on project management will not be repaid.

Therefore, we believe that the following sequence of actions can contribute significantly to the efficient and successful oversight management of software-intensive system acquisitions:

1. A formal analysis should be conducted to identify any common information generally needed by oversight management. Such information should be identified by source as either already available in the present form, easily derivable from available data, or requiring new data reporting channels together with integrity controls.
2. Experienced oversight managers should be surveyed to determine what information each has found most valuable and least valuable for decision purposes. Any information that can provide a cost/benefit relationship for judging the value of information exclusively prepared for oversight purposes should be solicited.
3. Experienced project managers should be surveyed to identify the questions most frequently asked, to examine how information needs were met, and to explore the use of automated tools in support of information needs for oversight management. Project

managers might also be asked to add items to the menu of oversight management methods presented in Sec. V.

4. Records should be kept of any techniques that appear to offer new insights for oversight managers; each of these techniques should be described in detail (see the Appendix for a sample).
5. Finally, AFSC might well be the clearinghouse to evaluate these ideas. For those that appear worthy, provisions should be made to test them in pilot operation on a suitable project. (Figure 4 indicates where applicable tools now exist and suggests gaps that could be the topics of future R&D efforts.)

For software-intensive projects, AFSC could insist on the following actions:

1. The oversight manager must inform project offices and key project personnel about his unique information needs. Project personnel are expected to fulfill the information needs of oversight management in a natural and efficient way, as a part of their job assignments. They should be encouraged to use automated tools and project management methods that provide visibility on subjects of importance to oversight managers, and to adjust their normal management style so as to be able to give necessary oversight information as a natural by-product.
2. Project management is expected to design its information system with the needs of oversight reviews in mind.

There are other aspects of project conduct that might also usefully support oversight management. For example, comprehensive *programmer work stations* (PWS) might be implemented to gather event statistics for oversight management and to track the progress that has been made on each module of each task in the overall project. The additional insight these statistics provide makes a PWS attractive, other things being equal.

Oversight management is clearly interested in major milestones in the overall schedule of events. In a conventional PERT chart, parallel paths may sometimes converge on a single milestone and then fan out

again to form subsequent parallel lines of endeavor. Obviously, such singular events are critical and the completion of each represents a major milestone. Except for such unique events, however, milestones may be difficult to recognize or to create automatically--a challenge similar to that of decomposing designs into program modules with automated processes. Perhaps some of these issues should be items for management R&D.

One of the most difficult challenges for the oversight manager is that of dealing with proposals from the project which save neither cost nor time, but which offer improvements to the quality of the product. A product quality metric is sorely needed in the computing community to facilitate such decisions. This is a researchable issue.

There are several program development methodologies in use today. But except for providing project-level information that just happens to help oversight management, none make explicit provision for the information needed by oversight management. A development methodology is a discipline that should produce better operational programs on budget and within schedule. Some of the popular programming development methodologies should be examined to determine where information offering management insights is naturally available, and to decide how it should be packaged for review by oversight managers. Given the prominence of the ADA language in the DoD, the ADA Programming Support Environment (APSE) is a particularly suitable candidate for such an extension. This is also a researchable issue.

Appendix
PROBABILISTIC BUDGETING: A PROMISING BUT UNTESTED METHOD

This appendix describes probabilistic budgeting, a possible oversight management tool that is illustrative of the ideas that could emerge from investigations such as those recommended in Sec. VI.

When the budget for a large development project is prepared, some fixed expenses, such as building depreciation, can be determined with great certainty. Other items, however, have to be designed and tested for feasibility, and therefore may be impossible to cost with any accuracy. For accounting purposes, dollar numbers must be attached to all items, so that total cost can be calculated; but by the time the total is generated, all the caveats about risk and the uncertainty of R&D schedules are likely to have been forgotten.

Totals are eventually presented to oversight management, with or without words of caution. As the funding process moves on, the totals become sacrosanct. Deletions and additions are then made, using the totals as a base. Cost tradeoffs based on these totals may either delight or confound the project staff, depending on the uncertainty built into the original individual cost esimates.

Data quality indicators have been proposed as a way for explicitly holding quality codes in a dynamic database.[1] A variant of this technique could improve the financial data provided for oversight managers.

Line managers could develop budgets in the usual way but assign a probability code to each expense item at the time the budget entries are submitted to the project office. Typical probability codes might be:

A. *100% certainty*: certain to happen as budgeted.
B. *Plus or minus 10%*: likely to happen as planned, unless some unforeseen event occurs, such as a key person quitting, airlines raising fares, or contract personnel being required to supplement staff.

---

[1] Robert L. Patrick, *Data Quality Indicators and Their Use in Data Base Systems,* The Rand Corporation, P-6491, May 1980.

C. *Plus or minus 25%*: significant uncertainty because of immature technology or the inexperience of the people undertaking the work.

D. *Plus or minus 50%*: budget figures highly unreliable because the work consists of research not yet reduced to common practice.

The computer support programs associated with the budget process can be modified to accept the probability codes, retain them in the budget database, and report them in various ways, e.g.,

- As the arithmetic grand total of expected expense, ignoring the probability codes (this is identical to today's process).
- As four independent subtotals, by probability code.
- As two virtual totals obtained by weighting and then adding or subtracting the independent subtotals to or from the expected grand total to get a budget range: The lower number would be the best-case result if all factors were in the project's favor; the higher number would be the worst-case result.

These new totals would provide oversight managers with best-case, likely, and worst-case budgets that could serve as guidelines for informed discussions of schedules, contingency funding, and parallel developments to reduce excessive risk.

FIL